Package 'brotli'

November 5, 2026

Type Package

Title A Compression Format Optimized for the Web

Version 1.4.0

Description A lossless compressed data format that uses a combination of the LZ77 algorithm and Huffman coding <https://www.rfc-editor.org/info/rfc7932/>. Brotli is similar in speed to deflate (gzip) but offers more dense compression.

License MIT + file LICENSE

URL https://jeroen.r-universe.dev/brotli

BugReports https://github.com/jeroen/brotli/issues

VignetteBuilder knitr

Suggests spelling, knitr, microbenchmark, rmarkdown, ggplot2

Language en-US

Config/roxygen2/version 8.0.0

NeedsCompilation yes

Author Jeroen Ooms [aut, cre] (ORCID: <https://orcid.org/0000-0002-4035-0289>), Google, Inc [aut, cph] (Brotli C++ library)

Maintainer Jeroen Ooms <jeroenooms@gmail.com>

Repository CRAN

Date/Publication 2026-11-05 11:40:09 UTC

Contents

brotli 2

Index **4**

brotli *Brotli Compression*

Description

Brotli is a compression algorithm optimized for the web, in particular small text documents.

Usage

```
brotli_compress(buf, quality = 11, window = 22)

brotli_decompress(buf)
```

Arguments

buf	raw vector with data to compress/decompress
quality	value between 0 and 11
window	log of window size

Details

Brotli decompression is at least as fast as for gzip while significantly improving the compression ratio. The price we pay is that compression is much slower than gzip. Brotli is therefore most effective for serving static content such as fonts and html pages.

For binary (non-text) data, the compression ratio of Brotli usually does not beat `bz2` or `xz (lzma)`, however decompression for these algorithms is too slow for browsers in e.g. mobile devices.

References

J. Alakuijala and Z. Szabadka (July 2016). *Brotli Compressed Data Format*. IETF Internet Draft https://www.rfc-editor.org/info/rfc9841/.

See Also

memCompress

Examples

```
# Simple example
myfile <- file.path(R.home(), "COPYING")
x <- readBin(myfile, raw(), file.info(myfile)$size)
y <- brotli_compress(x)
stopifnot(identical(x, brotli_decompress(y)))

# Compare to other algorithms
length(x)
length(brotli_compress(x))
length(memCompress(x, "gzip"))
```

```
length(memCompress(x, "bzip2"))
length(memCompress(x, "xz"))
```

Index

brotli, 2
brotli_compress *(*brotli*)*, 2
brotli_decompress *(*brotli*)*, 2

memCompress, *2*